1.计算字符串最后一个单词的长度，单词以空格隔开。

|  |
| --- |
| **import** java.util.Scanner;  **public** **class** Main {  public static void main(String[] args) {  Scanner sc = new Scanner(System.in);  String line = sc.nextLine();  String[] arr = line.split(" ");  System.out.println(arr[arr.length-1].length());  }  } |

2.写出一个程序，接受一个由字母、数字和空格组成的字符串，和一个字母，然后输出输入字符串中该字母的出现次数。不区分大小写。

|  |
| --- |
| **import** java.util.Scanner;  **public** **class** Main {  **public** **static** **void** main(String[] args) {  **int** count = 0;  Scanner sc = **new** Scanner(System.***in***);  String input = sc.nextLine().toLowerCase();  **char** tag = sc.nextLine().toLowerCase().toCharArray()[0];  **for** (**int** i = 0; i < input.length(); i++) {  **if** (input.charAt(i) == tag) {  count++;  }  }  System.***out***.println(count);  }  } |

1. 计算机随机生成1-1000内的随机数（N<=1000），其中重复的数字只保留以一个，其他去掉，并按照从小到大顺序排序。

|  |
| --- |
| **import** java.util.Scanner;  **import** java.util.TreeSet;  **public** **class** main {  **public** **static** **void** main(String[] args) {  Scanner sc = **new** Scanner(System.***in***);  **while** (sc.hasNext()) {  TreeSet<Integer> set = **new** TreeSet<Integer>();  **int** n = sc.nextInt();  **if** (n > 0) {  **for** (**int** i = 0; i < n; i++) {  set.add(sc.nextInt());  }  }  **for** (Integer i : set) {  System.***out***.println(i);  }  }  }  } |

1. 字符串排序：给定n个字符串，请对n个字符串按照字典序排列。

|  |
| --- |
| import java.util.\*;  public class Main {  public static void main(String[] args) {  Scanner sc = new Scanner(System.in);  while (sc.hasNext()){  int n = sc.nextInt();  String [] arr = new String[n];  for(int i=0;i<n;i++){  String str = sc.next();  arr[i] = str;  }  Arrays.sort(arr);  for(int i=0;i<arr.length;i++){  System.out.println(arr[i]);  }  }  sc.close();  }  } |

1. 词语逆向：接受一个只包含小写字母的字符串，然后输出该字符串反转后的字符串。（字符串长度不超过1000）

|  |
| --- |
| **import** java.util.Scanner;  **public** **class** Main {  **public** **static** **void** main(String[] args) {          Scanner scanner = **new** Scanner(System.in);  **while**(scanner.hasNext()){              String str= scanner.nextLine();              //将字符串变成char数组再倒序输出  **char**[] chars= str.toCharArray();  **for**(**int** i=chars.length-1;i>=0;i--){                  System.out.print(chars[i]);              }              System.out.println();          }      }  } |

1. 句子逆向：将一个英文语句以单词为单位逆序排放。例如“I am a boy”，逆序排放后为“boy a am I”  
   所有单词之间用一个空格隔开，语句中除了英文字母外，不再包含其他字符。

|  |
| --- |
| **public** **class** ReverseWord {  **public** **static** **void** main(String[] args) {          Scanner scanner = **new** Scanner(System.in);          Deque<String> words = **new** LinkedList<>();  **for** (String word : scanner.nextLine().split("\\s+")) {              words.addFirst(word);          }          System.out.println(String.join(" ", words));      }  } |

1. 寻找字符：输入一个int型的正整数，计算出该int型数据在内存中存储时1的个数。

注意：这里时在内存中存储时，意思就是二进制格式中有几个1。

在Integer类中有静态方法：

toBinaryString（int i）方法，此方法返回int变量的二进制表示的字符串。

toHexString（int i）方法，返回int变量的16进制表示的字符串。

toOctalString（int i）方法,返回int变量的8进制表示字符串表示的字符串。

|  |
| --- |
| **import** java.util.Scanner;  **public** **class** Main{  **public** **static** **void** main(String[] args){          Scanner sc = **new** Scanner(System.in);  **while**(sc.hasNext()){  **int** num = sc.nextInt();              //计算1的个数  **int** count = 0;              String toBinary = Integer.toBinaryString(num);  **for**(**int** i = 0; i < toBinary.length(); i++){  **char** c = toBinary.charAt(i);  **if**(c == '1'){                      count++;                  }              }              System.out.println(count);          }      }  } |

8、计算重复次数：编写一个函数，计算字符串中含有的不同字符的个数。字符在ACSII码范围内(0~127)，换行表示结束符，不算在字符里。不在范围内的不作统计。多个相同的字符只计算一次

例如，对于字符串abaca而言，有a、b、c三种不同的字符，因此输出3。

|  |
| --- |
| import java.util.HashSet; import java.util.Scanner;  public static void main(String[] args) {  Scanner sc = new Scanner(System.in);  while (sc.hasNext()){       char[] next = sc.next().toCharArray();       HashSet<Integer> set = new HashSet<>();  **for** (**int** i = 0; i < next.length; i++) {           //(int)next.toCharArray()[i]  **int** ascValue = (**int**)next[i];  **if**(ascValue>0 && ascValue<127){              set.add(ascValue);            }        }        System.out.println(set.size());  }  }  } |

9、求质因子：功能:输入一个正整数，按照从小到大的顺序输出它的所有质因子（重复的也要列举）（如180的质因子为2 2 3 3 5 ）最后一个数后面也要有空格。

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** main(String[] args){

        Scanner scan = **new** Scanner(System.in);

**long** num = Long.parseLong(scan.next());

        getPrimer(num);

    }

**public** **static** **void** getPrimer(**long** num){

**for** (**int** i= 2;i <= num; i++){

**if** (num % i==0){

                System.out.print(i + " ");

                getPrimer(num/i);

**break**;

            }

**if** (i==num){

                System.out.print( i + "");

            }

        }

    }

}